Distributed agile: our journey and guiding principles
Summary

Agile software development and the breed of agile methodologies (XP, SCRUM, DSDM, etc.) have gained popularity since 2001. Agile methodologies were primarily founded for software projects executed at a single location. Today, with many adopters and practitioners across the globe, agile methodologies are showing promising results in multi-site projects too. Offshore delivery models have been successful in application maintenance and enhancement projects for more than two decades. In the case of development projects, iterative lifecycle approaches are more widespread and acceptable than the classical waterfall approach in delivering results and ensuring customer satisfaction. Distributed agile software development involves software projects done by agile teams located across geographies. This paper presents Mindtree's journey and learnings on distributed agile and offshoring.

Contents

What is distributed agile? ................................................................. 03
What do we do? .................................................................................. 03
Distributed agile at Mindtree: the guiding principles .......... 04
Our capability ................................................................................... 05
What next? ....................................................................................... 05
What is distributed agile?
Agile is all about delivering business value in short iterations at a sustainable pace, adapting to changing business needs. Agile software development focuses on early delivery of working software and considers working software as the primary measure of progress. It creates an environment that responds to change by being flexible and nimble. It discourages creation of extensive documents that do not add any value. Distributed agile software development and testing is simply applying agile principles and practices to software projects executed by distributed teams or teams located at different sites. These could be at two or more floors of the same building, different buildings, cities or countries across geographies and time zones.

What do we do?
Mindtree started implementing agile methodologies in geographically distributed teams as early as 2002. In adopting distributed agile we believe in five aspects: appreciation, readiness, capability, experience and methodology, as shown in fig. 01.

Appreciation: Mindtree understands and appreciates the significance of Global Software Engineering (GSE) and agile. Agile has been the mantra of success and the adoption of distributed agile continues to rise. We study current industry trends and work with eminent thought leaders to serve our customers better.

Readiness: We value the vital role played by physical infrastructure such as creative workspaces, collaboration tools and communication networks. We focus consistently on sustaining and upgrading our readiness to nurture geographically distributed teams.

Capability: We are relentlessly committed to competency building and specialization in areas such as distributed agile. Our investment in knowledge management and thought leadership has helped us build strong distributed agile capabilities over the past several years.

Experience: We have executed numerous projects that involved GSE and agile. We continue to work with many of our customers in these areas.

Fig. 01: Distributed agile at Mindtree:
The five aspects
**Methodology:** Our focus on the first four aspects has helped us work with our customers in defining context-specific methodologies with a good balance of management and engineering practices.

Through this experience we have identified ten guiding principles that help us deliver success through distributed agile.

**Distributed agile at Mindtree: the guiding principles**

Simply put, adherence to the Agile Manifesto and Agile Principles is the foundation of agile teams. In addition to the 12 principles declared by the founders of the agile manifesto, we value the following ten guiding principles:

1. **Methodology is driven by project teams:** Agile software development in a distributed environment does not mean step-by-step implementation of any specific agile methodology such as Scrum, with high expectations on timely, high-quality delivery. It means collaboration amongst distributed teams to collate processes that follow agile principles and to put together a methodology that works for them. Projects that follow distributed agile suffer when a methodology accepted by a sub team drives the rest of the team. Successful distributed agile projects happen because of collaborative teams that drive to define a methodology for themselves. The definition of such a methodology happens by means of open communication and minor adjustments to make things work as expected. In other words, methodology is driven by project teams. More importantly, a methodology that works for one distributed ecosystem may not work for another distributed ecosystem. This is because, for any methodology, while the basic tenets remain intact, the implementation details vary across ecosystems. Hence, methodology is an internal affair of the project team.

2. **Consistent usage of common tools improves productivity:** Team members in a distributed team must have access to a standardized set of tools for engineering activities such as design, coding, static analysis, unit testing, build automation, test automation, defect tracking and so on. Besides, they need to use such tools consistently in order to realize the benefits. Disparate tools result in compatibility issues and impact team productivity.

3. **Infrastructure for communication and coordination is crucial:** Team members of geographically distributed teams depend on phone calls, chat, email and video conferencing for communication. Also, they depend on web-based tools for agile project management, issue tracking, defect tracking, etc. It is crucial to have an infrastructure that supports distributed development in order to relieve teams from technical issues related to communication and coordination.

4. **Knowledge management is the key to success:** Assimilation, creation, dissemination and regular upkeep of knowledge related to technology as well as domain elements of a project are critical to the success of the project. This is especially important in distributed agile projects. This is because teams have to focus consistently on delivering working software over short iterations at a sustainable pace and respond to changes coming from business users at the same time. Systematic and consistent focus on knowledge management improves our ability to induct new joiners in order to expand teams and also to manage attrition effectively.

5. **Quality is multi-dimensional and owned by everybody:** Quality can be seen in terms of intrinsic or internal quality and external quality. External quality is an attribute that relates to the end-user experience. It can be assessed and improved through black box testing and defect prevention. Internal quality is visible to various groups in the development team, such as designers, developers, maintainers and technical reviewers. Internal quality is invisible to end-users. It can be assessed and improved through reviews and static analysis. Internal quality can be improved by defect prevention, as well as defect detection, followed by analysis and correction or defect fixing.

Quality can be improved from different dimensions or streams of activities, such as inspection of requirements, design reviews, testing of functions, performance, security and compliance, exploratory testing, etc. Agile teams understand this multi-dimensional aspect of quality and value the whole team approach. Obviously, the set of metrics or measures used to understand progress needs to be multi-dimensional. A single dimensional view of quality is simply not comprehensive enough. Also, in distributed teams, team members from every location have to demonstrate a relentless focus on quality.
6. **Distributed agile requires an inclusive approach:**
Distributed agile teams need to consider an inclusive approach in order to nurture distributed ecosystems. This is more important than the distribution of functional modules or user stories across sites. Facilitating a base camp at a central location at the beginning of the project and having an adequate travel budget for team members to travel across sites at regular intervals is the first step in ensuring inclusion. Implementing distributed test drives or reviews, distributed retrospectives and distributed root cause analysis are also ways to nurture inclusion.

7. **Governance is the backbone of successful distributed teams:** Geographically distributed teams cannot function on their own. Collaborative governance is critical to the success of projects executed by distributed teams.

8. **Automation enables sustainable pace:** Automation of engineering tasks such as build creation, test data creation, unit test execution, regression testing, test result analysis, etc., is necessary to optimize or avoid manual efforts spent on routine tasks. With automation, team members get adequate time to focus on critical tasks that need manual intervention. The significance of automation in distributed teams is greater than that of collocated teams.

9. **It is essential to streamline the accumulation and pay-off of technical debt:** Distributed teams need to be aware, aligned and organized in managing technical debt in order to deliver maintainable, robust software.

10. **Iteration progression is a common phenomenon and ensuring early success is a collective responsibility:** Distributed teams progress over the first three or four iterations. Aiming for instantaneous results from the first iteration is unrealistic. Ensuring early success is a collaborative responsibility of both project and governance teams.

**Our capability**
We have ten years of proven experience in agile methodologies. We have executed projects ranging from software development and maintenance to testing, using agile methodologies with distributed teams as well as collocated offshore teams. Our experience includes:
- 50,00,000+ man hours of experience in agile projects
- 1000+ agile team members
- 100+ ongoing agile projects

We understand agile as well as other evolutionary methodologies. Our experience in executing projects in onsite-offshore models enables us to apply agile principles and best practices in distributed teams. To make this work at an organizational level, we have created subject matter experts and agile coaches. We also facilitate training programs on distributed agile to build agile capabilities in our organization. These are the crucial steps we adopt to make it work.

We promote agile through Mindtree Agile Council and Agile Community. Mindtree Agile Council, a team of agile experts, focuses on nurturing agile capabilities at Mindtree and facilitating competency building programs. Agile Community at Mindtree is a knowledge management community that provides a common platform for all practitioners and promotes knowledge sharing. In addition to these we contribute to external conferences and online media by sharing our success stories, presenting white papers and participating in discussions.

---

**What next?**
Mindtree has contributed to several success stories on distributed agile. Every year we see many new engagements kicking off in this model. We have experienced the symbiotic relationship between distributed agile and offshoring and so have our customers. We believe that distributed agile will be increasingly adopted over the coming years. For more information on Mindtree distributed agile capabilities or case studies please contact raja_bavani@mindtree.com.

**References:**
Mindtree distributed agile blogs:
http://www.blogs.mindtree.com/author/raja-bavani

Mindtree articles and white papers:
http://www.mindtree.com/services/agile
About the author:
Raja Bavani is Chief Architect of Mindtree’s Product Engineering Services (PES) and IT Services (ITS) groups and plays the role of agile evangelist. He has more than 20 years of experience in the IT industry and has published papers at international conferences on topics related to code quality, distributed agile, customer value management and software estimation. He is a member of IEEE and IEEE computer society. He regularly interfaces with educational institutions to offer guest lectures and writes for technical conferences. He also writes for magazines such as Agile Record, Cutter IT Journal, IEEE Software and SD Times.

About Mindtree
Mindtree is a global information technology solutions company with revenues of over USD 400 million. Our team of 11,000 experts engineer meaningful technology solutions to help businesses and societies flourish. We enable our customers achieve competitive advantage through flexible and global delivery models, agile methodologies and expert frameworks.